

ABOUT INVERSE 3-SAT

XAVIER LABOUZE

ABSTRACT. The Inverse 3-SAT problem is known to be coNP Complete: Given ϕ a set of models on n variables, is there a 3-CNF formula such that ϕ is its exact set of models ? An immediate candidate formula F_ϕ^3 arises, which is the conjunction of all 3-clauses satisfied by all models of ϕ . The (co)Inverse 3-SAT problem can then be resumed: Given ϕ a set of models on n variables, is there a model of F_ϕ^3 which is not in ϕ ?

This article uses two important intermediate results: 1- The candidate formula can be transformed in polynomial time into an equivalent formula F_ϕ which is 3-closed under resolution. A crucial property of F_ϕ is that the induced formula $F_{\phi|I}$ by applying any partial assignment I of the n variables to F_ϕ is unsatisfiable iff its 3-closure contains the empty clause. 2- A set of partial assignments (of polynomial size) which subsume all assignments not in ϕ can be computed in polynomial time.

So, the (co)Inverse 3-SAT question is equivalent to decide whether it exists a partial assignment I not in ϕ such that the 3-closure of $F_{\phi|I}$ does not contain the empty clause, and the problem is solvable in polynomial time.

1. INTRODUCTION

The satisfiability problem has been one of the most studied problems in computational complexity [1, 2, 3, 4, 6, 7, 9]. Kavvadias and Sideri have shown that the Inverse 3-SAT problem is coNP Complete [5]: Given ϕ a set of models on n variables, is there a 3-CNF formula such that ϕ is its exact set of models ? An immediate candidate 3-CNF formula F_ϕ^3 arises which is the set of all 3-clauses satisfied by all models in ϕ . Since F_ϕ^3 is the most restricted 3-CNF formula (in term of its model set) which is satisfied by all models in ϕ , the (co)Inverse 3-SAT problem can then be defined: Given ϕ a set of models on n variables, is there a model of F_ϕ^3 which is not in ϕ ? The properties of F_ϕ^3 will bring a new interesting way to solve the Inverse 3-SAT problem.

In the second part of the article, all the needed notations will be defined. In the third part, the main ideas of the algorithm presented in the fourth part will be developed.

2. PRELIMINARIES

2.1. 3-CNF formula. A CNF propositionnal formula F is regarded in the standard way as a set of clauses, where each clause is regarded as a set of literals, and each literal as a boolean variable or its negation. Whether x is a positive or a negative literal, \bar{x} denotes its complement. The size of a set A (denoted $|A|$) is the number of its elements. The clause $c = \{x, y, z\}$ is denoted (xyz) . $c \setminus \{x\}$ is the clause (yz) . The empty clause, denoted (\emptyset) , is equivalent to *false*. A 3-CNF formula is a CNF formula with at least one clause of size 3 (the other clauses are of size 3 or smaller).

2.2. Assignment. Let F be a 3-CNF formula on n variables $\{x_1, x_2 \dots x_n\}$. Each variable x_i can be assigned to the value v_i . A (total) *assignment* of the n variables is a set of n values $\{v_1, v_2 \dots v_n\}$, where the value v_i is assigned to the variable x_i . A value v is equal to 0 (*false*) or 1 (*true*), the opposite of the value v , $\bar{v} = 1 - v$. A clause of F is satisfied when at least one of its literals is set (assigned) to *true*. F is satisfiable if it exists a truth assignment of the n variables which satisfies all its clauses. Such a truth assignment is called a *model*. A *partial* assignment on k variables is the subset of a total assignment restricted to the values of the chosen k variables ($k \leq n$).

Definition 2.1. Given F a 3-CNF formula on n variables $\{x_1, x_2 \dots x_n\}$; c , a clause in F ; I , a partial assignment of k variables among (x_i) ($k \leq n$).

- (1) Let $F|_I$ be the induced formula by applying I to F : Any clause that contains a literal which evaluates to *true* under I is deleted from the formula and any literals that evaluate to *false* under I are deleted from all clauses - the clauses that become empty by this deletion remain in the formula as the empty clause.
- (2) Let $c|_I$ be the induced clause by applying I to c : If c contains a literal which evaluates to *true* under I then $c|_I = \text{true}$; If c contains a subset A of literals all set to *false* under I then $c|_I = c \setminus A$; If c does not contain any literal set by I then $c|_I = c$.

2.3. Subsumption. A clause c is said to *subsume* a clause d , and d is *subsumed* by c , if the literals of c are a subset of those of d (each clause subsumes itself then). A (partial) assignment I is said to *subsume* a (partial) assignment J , and J is *subsumed* by I , if the values of I are a subset of those of J . Given ψ a certain set of (partial) assignments, a (partial) assignment $I \in \psi$ if it subsumes a (partial) assignment of ψ .

2.4. Resolution. Two clauses, $c_1 = (Ax)$ and $c_2 = (B\bar{x})$, can be resolved in a third clause $c = (AB)$, so called *resolvent*, where A and B are two subsets of literals. A CNF formula F is said to be *closed under resolution* (or just closed) if no clause of F is subsumed by a different clause of F , and the resolvent of each pair of resolvable clauses is subsumed by some clause of F . The *closure* of a CNF formula F is the CNF formula (denoted F^c) that derived from F by a series of resolution (which add clauses) and subsumptions (which delete clauses), and is closed (it is easy to see that closure under resolution is unique) [10].

F^c can be separate into 2 disjoint subsets: $F^c = 3-F^c \cup F^r$, where $3-F^c$ is the *3-closure* of F , i. e. the subset of F^c containing only clauses of size 3 or less, and F^r contains clauses of size 4 or more.

3. DISCUSSION BEFORE THE ALGORITHM

Given $\phi = \{m_1, m_2 \dots m_{|\phi|}\}$, a set of $|\phi|$ models on n variables $(x_i)_{i \leq 1 \leq n}$ (an element of ϕ will be called either assignment or model or simply element according to the context). Let F_ϕ^3 the set of all 3-clauses satisfied by all models in ϕ .

3.1. The 3-closure of F_ϕ^3 can be computed in polynomial time.

Proof. Since F_ϕ^3 contains all 3-clauses satisfied by all models in ϕ , all possible 3-clauses resulting from the resolution of clauses implied by F_ϕ^3 are subsumed by some clause in F_ϕ^3 . Since any resolvent of size 2 or less results from the resolution of clauses of size 3 or less, the 3-closure under resolution of F_ϕ^3 can be computed in polynomial time. □

Notation. Call F_ϕ (or F if it is not confusing) the 3-closure of F_ϕ^3 .

Proposition 3.1. Given I , a partial assignment of k variables among (x_i) ($k \leq n$), the 3-closure of $F_{\phi|I}$ is computable in polynomial time.

Proof. Directly □

Remark. (1) Each clause of F_ϕ^3 is subsumed by a clause in F_ϕ and F_ϕ is equivalent to F_ϕ^3 . (2) As $F_\phi^c = F_\phi \cup F_\phi^r$ then all clauses of F_ϕ^r result from resolution of clauses of F_ϕ^3 or some iterated resolvents of clauses of F_ϕ^3 .

Example 1. Take $n = 5$ and 8 models $(m_i)_{1 \leq i \leq 8}$ in ϕ .

$\phi = \{00111, 01011, 10101, 11100, 11111, 10011, 01101, 00100\}$

By gathering all the 3-clauses satisfied by all the models of ϕ :

$$\begin{aligned} F_\phi^3 = & (x_1x_2x_3)(\bar{x}_1\bar{x}_2x_3)(x_1\bar{x}_2x_5)(\bar{x}_1x_2x_5)(x_1x_3x_4)(\bar{x}_1x_3x_4)(x_1x_3x_5)(\bar{x}_1x_3x_5)(x_1\bar{x}_4x_5) \\ & (\bar{x}_1\bar{x}_4x_5)(x_2x_3x_4)(\bar{x}_2x_3x_4)(x_2x_3x_5)(\bar{x}_2x_3x_5)(x_2\bar{x}_4x_5)(\bar{x}_2\bar{x}_4x_5)(x_3x_4x_5)(x_3x_4\bar{x}_5) \\ & (x_3\bar{x}_4x_5)(\bar{x}_3\bar{x}_4x_5) \end{aligned}$$

Its 3-closure is:

$$F_\phi = (x_1x_2x_3)(\bar{x}_1\bar{x}_2x_3)(x_1\bar{x}_2x_5)(\bar{x}_1x_2x_5)(x_3x_4)(x_3x_5)(\bar{x}_4x_5)$$

3.2. $F_{\phi|I}$ is unsatisfiable iff its 3-closure contains the empty clause. Given I , a partial assignment of k variables among (x_i) ($k \leq n$).

Proposition 3.2. Given F , a 3-CNF formula on n variables $(x_i)_{i \leq 1 \leq n}$. F is closed under resolution implies $F|_I$ is closed under resolution.

Proof. If $c_1 \ni x_i$ and $c_2 \ni \bar{x}_i$ are in $F|_I$ (in particular, x_i is unset by I), pick clauses d_1, d_2 in F which restrict to c_1 and c_2 , respectively. Then $x_i \in d_1$ and $\bar{x}_i \in d_2$, hence their resolvent $(d_1 \setminus \{x_i\}) \cup (d_2 \setminus \{\bar{x}_i\})$ is subsumed by some $d \in F$. If d contains a literal made *true* under I , then so does d_1 or d_2 , contradicting their choice. Thus, $d|_I$ is in $F|_I$, and it subsumes $(c_1 \setminus \{x_i\}) \cup (c_2 \setminus \{\bar{x}_i\})$.

Thanks to Emil Jeřábek (<http://cstheory.stackexchange.com/a/16835/6346>). □

Proposition 3.3. $F_{\phi|I}$ is unsatisfiable iff its 3-closure contains the empty clause.

Proof. As $F_\phi^c = F_\phi \cup F_\phi^r$ then $F_{\phi|I}^c = F_{\phi|I} \cup F_{\phi|I}^r$. Suppose the 3-closure of $F_{\phi|I}$ is unsatisfiable (the other implication is obvious). Then $F_{\phi|I}^c$ is unsatisfiable and it contains the empty clause (from the previous proposition and the Quine's theorem [8]: A formula closed under resolution is unsatisfiable iff it contains the empty clause).

1- As F_ϕ^c is equivalent to F_ϕ then $F_{\phi|I}^c$ is equivalent to $F_{\phi|I}$ 2- Two equivalent formulas have the same 3-closure. 3- If the empty clause is in a formula then it is in its 3-closure (since $|\emptyset| = 0$). Hence (\emptyset) is in the 3-closure of $F_{\phi|I}$. □

3.3. A set of partial assignments subsuming all assignments not contained in ϕ can be computed in polynomial time. Consider some total order among the n variables, say the lexicographic one.

Definition 3.1. Some additional useful definitions:

- (1) Let M_k be the set of all 2^k partial assignments (I_k) on the first k values of the variables ($1 \leq k \leq n$).
- (2) Let $\phi_k = \{I_k \in M_k / I_k \in \phi\}$
- (3) Let $\bar{\phi}_k = \{I_k \in M_k / I_{k-1} \in \phi_{k-1} \text{ and } I_k \notin \phi_k\}$ ($I_0 = \emptyset$ and ϕ_0 is the empty set)
- (4) Let $\bar{\phi} = \bigcup_k \bar{\phi}_k$

- (5) Let $m_{i,j}$ the restriction of $m_i \in \phi$ to its first j values and $\bar{m}_{i,j}$ the restriction of $m_i \in \phi$ to its first $j - 1$ values ($j \geq 1$) concatenated with the opposite of its j^{th} value (as last value).

Proposition 3.4. About $\bar{\phi}_k$

- (1) The extension to the rest of the n variables of any partial assignment of $\bar{\phi}_k$ is not in ϕ .
- (2) An assignment I_n of the n variables does not belong to ϕ iff $\exists k \leq n$, $I_k \in \bar{\phi}_k$ where I_k is the partial assignment issued from I_n restricted to the first k values.
- (3) The computation of $\bar{\phi}_k$ can be done in polynomial time.

Proof. (1) Since any element of $\bar{\phi}_k$ is not in ϕ , neither is any extension of it.
(2) If $I_n \notin \phi$ then obviously $\exists k \leq n$, $I_k \in \bar{\phi}_k$. If $\exists k \leq n$, $I_k \in \bar{\phi}_k$ where I_k is the partial assignment issued from I_n restricted to the first k values then by (1) any extension of $I_k \notin \phi$ and $I_n \notin \phi$.
(3) $|\phi_k|, |\bar{\phi}_k| \leq |\phi|$ (and $|\bar{\phi}| \leq n|\phi|$). The computation of ϕ_k can obviously be done in polynomial time. So can be the computation of $\bar{\phi}_k$: for each model $m_i \in \phi$, compute $\bar{m}_{i,k}$, put it in $\bar{\phi}_k$ if it does not belong to ϕ_k . □

Proposition 3.5. About $\bar{\phi}$

- (1) The extension to the rest of the n variables of any partial assignment of $\bar{\phi}$ is not in ϕ .
- (2) $\bar{\phi}$ is a set of partial assignments subsuming all assignments of the n variables which are not in ϕ ($|\bar{\phi}| \leq n|\phi|$).
- (3) $\bar{\phi}$ can be computed in polynomial time.

Proof. Directly from the previous proposition and the definition of $\bar{\phi}$. □

Remark. As we are interested in partial assignments which could be extended to an entire model for the 3-CNF F , we can only consider the $\bar{\phi}_k$ sets for $k > 3$ without changing anything to the rest.

Example 2. Take $n = 5$ and 8 models $(m_i)_{1 \leq i \leq 8}$ in ϕ .

$\phi = \{00111, 01011, 10101, 11100, 11111, 10011, 01101, 00100\}$ (as Example 1)

The 3-closure of the candidate formula has been established:

$F_\phi = (x_1 x_2 x_3)(\bar{x}_1 \bar{x}_2 x_3)(x_1 \bar{x}_2 x_5)(\bar{x}_1 x_2 x_5)(x_3 x_4)(x_3 x_5)(\bar{x}_4 x_5)$

Let build the sets $(\bar{\phi})_k$ for $4 \leq k \leq n(= 5)$

- $k = 4$
 $\phi_4 = \{0011, 0101, 1010, 1110, 1111, 1001, 0110, 0010\}$
 $\bar{m}_{1,4} = 0010 \in \phi_4$ ($= m_{8,4}$ so $\bar{m}_{8,4} = m_{1,4} \in \phi_4$)
 $\bar{m}_{2,4} = 0100 \notin \phi_4$ ($\in \bar{\phi}_4$)
and so on until $\bar{\phi}_4 = \{0100, 1011, 1000, 0111\}$
- $k = 5$
In the same way, $\bar{\phi}_5 = \{00110, 01010, 10100, 11101, 11110, 10010, 01100, 00101\}$

Hence $\bar{\phi} = \bar{\phi}_4 \cup \bar{\phi}_5$

3.4. Another formulation of the question: Is there a partial assignment $I \in \bar{\phi}$ such that the 3-closure of $F_{\phi|I}$ does not contain the empty clause ?

Proposition 3.6. The (co)Inverse 3-SAT question "Is there a model of F_ϕ^3 which is not in ϕ ?" is equivalent to the question "Is there a partial assignment $I \in \bar{\phi}$ such that the 3-closure of $F_{\phi|I}$ does not contain the empty clause ?"

Proof. If it exists a partial assignment $I \in \bar{\phi}$ such that the 3-closure of $F_{\phi|I}$ does not contain the empty clause then :

- 1) All extensions of I on the rest of the n variables are not in ϕ (from Prop. 3.4).
- 2) $F_{\phi|I}$ is satisfiable (from Prop. 3.2).

Then I extended (concatenated) with a model of $F_{\phi|I}$ is a model of F_{ϕ}^3 which is not in ϕ .

If it exists m , a model of F_{ϕ}^3 (m is also a model F_{ϕ}), which is not in ϕ then it exists a partial assignment $I_m \in \bar{\phi}$ which subsumes m (since $\bar{\phi}$ is a set of partial assignments which subsume all assignment not in ϕ). Then $F_{\phi|I_m}$ is satisfiable (if not, no extension of I_m can satisfies neither F_{ϕ} nor F_{ϕ}^3 : contradiction) and its 3-closure does not contain the empty clause. □

4. THE ALGORITHM

Input. ϕ , a set of models over n variables.

Step 1. Compute F_{ϕ} , the 3-closure of the candidate formula.

Step 2. Compute $\bar{\phi}$, a set of partial assignments subsuming all assignments $\notin \phi$.

Step 3. For each partial assignment $I \in \bar{\phi}$, compute the 3-closure of $F_{\phi|I}$ and check whether it contains the empty clause.

Output. Yes or No, answering the question: Is there a partial assignment $I \in \bar{\phi}$ such that the 3-closure of $F_{\phi|I}$ does not contain the empty clause ?

Proposition 4.1. This algorithm lets solve the (co)Inverse 3-SAT problem. Each step can be computed in polynomial time without explosion in size.

Proof. This algorithm obviously finishes. It outputs the answer to the question: Is there a partial assignment $I \in \bar{\phi}$ such that the 3-closure of $F_{\phi|I}$ does not contain the empty clause ? which is equivalent to the classical (co)Inverse 3-SAT question. Its polynomial-time computation comes directly from the previous results of the article. □

Example 3. Take $n = 5$ and 8 models $(m_i)_{1 \leq i \leq 8}$ in ϕ .

$\phi = \{00111, 01011, 10101, 11100, 11111, 10011, 01101, 00100\}$ (as Example 1 and 2) F_{ϕ} and $\bar{\phi}$ have been found:

$$F_{\phi} = (x_1 x_2 x_3)(\bar{x}_1 \bar{x}_2 x_3)(x_1 \bar{x}_2 x_5)(\bar{x}_1 x_2 x_5)(x_3 x_4)(x_3 x_5)(\bar{x}_4 x_5)$$

$$\bar{\phi} = \{0100, 1011, 1000, 0111, 00110, 01010, 10100, 11101, 11110, 10010, 01100, 00101\}$$

$F_{0100} = (\emptyset)$ but $F_{1011} = (x_5)$ so the candidate formula has at least one model $m \notin \phi$ ($m = 10111$).

5. CONCLUSION

The (co)Inverse 3-SAT problem can be solved in polynomial time.

REFERENCES

1. Stephen A. Cook, *The complexity of theorem-proving procedures*, Proceedings of the third annual ACM symposium on Theory of computing (New York, NY, USA), STOC '71, ACM, 1971, pp. 151–158.
2. Michael R. Garey and David S. Johnson, *Computers and intractability: A guide to the theory of NP-completeness*, W. H. Freeman & Co., 1979.

3. Jun Gu, Paul W. Purdom, John Franco, and Benjamin W. Wah, *Algorithms for the satisfiability (SAT) problem: A survey*, Satisfiability Problem: Theory and applications (Ding-Zhu Du, Jun Gu, and Panos Pardalos, eds.), DIMACS Series in Discrete Mathematics and Theoretical Computer Science, American Mathematical Society, 1997, pp. 19–152.
4. Richard M. Karp, *Reducibility among combinatorial problems*, Complexity of Computer Computations (R. E. Miller and J. W. Thatcher, eds.), Plenum Press, 1972, pp. 85–103.
5. Dimitris Kavvadias and Martha Sideri, *The inverse satisfiability problem*, Computing and Combinatorics (Jin-Yi Cai and ChakKuen Wong, eds.), Lecture Notes in Computer Science, vol. 1090, Springer Berlin Heidelberg, 1996, pp. 250–259.
6. Leonid A. Levin, *Universal sequential search problems*, Problems of Information Transmission **9** (1973), no. 3, 265–266.
7. Christos H. Papadimitriou, *Computational complexity*, Addison-Wesley, 1994.
8. W. Quine, *The problem of simplifying truth functions*, American Mathematical Monthly **59** (1952), no. 8, 521–531.
9. Leslie G. Valiant, *The complexity of enumeration and reliability problems*, SIAM Journal on Computing **8** (1979), no. 3, 410–421.
10. Allen VanGelder and Yumi K. Tsuji, *Satisfiability testing with more reasoning and less guessing*, Tech. report, Santa Cruz, CA, USA, 1995.

DÉPARTEMENT DE MATHÉMATIQUES, UNIVERSITÉ PARIS 11, FRANCE

Current address: Faculté de Pharmacie, F-92296 Châtenay-Malabry Cedex, France

E-mail address: `xavier.labouze@u-psud.fr`